Express Mail Label No. EL 977712466 US

[0001] CHANNEL GAIN ESTIMATION IN A RAKE RECEIVER USING COMPLEX WEIGHT GENERATION (CWG) ALGORITHMS

[0002] CROSS REFERENCE TO RELATED APPLICATION

[0003] This application claims priority from U.S. provisional application No. 60/428,413, filed November 22, 2002, which is incorporated by reference as if fully set forth.

[0004] FIELD OF INVENTION

[0005] The present invention relates to wireless communication systems. More specifically, the present invention relates to the filtering of communications using a Complex Weight Generation (CWG) algorithm.

[0006] BACKGROUND OF THE INVENTION

[0007] A typical wireless communication system, such as specified in the Third Generation Partnership Project (3GPP), transmits downlink communications from a base station to one or a plurality of wireless transmit/receive units (WTRUs). An uplink communication occurs when the WTRU transmits to the Base Station (BS).

[0008] In a direct sequence Code Division Multiple Access (CDMA) transmission system, data is modulated by spreading it into a wideband radio frequency signal using a spreading code sequence. A communication system assigns different spreading codes to each user enabling them to communicate using the same radio frequency band. Receivers operate by correlating or despreading the received signal with a known spreading code sequence.

[0009] A receiver may receive time offset copies of a transmitted communication signal known as multi-path fading. The signal energy is dispersed over time due to distinct multi-paths and scattering. If the receiver has some information about the channel profile, the receiver may estimate the communication signal by combining the multi-path copies of the signal to improve performance. For example, one such method

gathers signal energy by assigning correlator branches to different paths and combining their outputs constructively.

[0010] In a CDMA system, a Rake receiver is conventionally used. As shown in Figure 1, a Rake receiver 10 consists of a bank of "sub-receivers" 20_A, 20_B...20_N and a combiner 30. Each "sub-receiver" 20 constitutes a Rake finger, i.e., multipath, which includes a delay 25_A, 25_B...25_N, a despreader 35_A, 35_B...35_N, a complex weight generator 45_A, 45_B...45_N and a demodulator (or multiplier) 55_A, 55_B...55_N, where the complex weight generator 45 estimates the channel gain. The channel gain is a complex parameter representing the amplitude attenuation and phase rotation of the signal received via antenna 60 and the sub-receivers 20. The demodulator (or multiplier) 55 is essentially a multiplier that multiplies the output of the despreader 35 with the complex weight provided by complex weight generator 45, whereby the output of multiplier 55 is a phase-rotation-removed and amplitude-weighted despread signal. Therefore, the combiner 30 coherently (or in co-phase) combines all signals received from all of the "sub-receivers" 20.

[0011] The Rake receiver 10 has several "fingers," one for each path. In each finger, a path delay with respect to some reference delay, such as the direct or the earliest received path, must be estimated and tracked throughout the transmission. Rake receivers are able to exploit multi-path propagation to benefit from path diversity of the transmitted signal. Using a plurality of paths, or rays, increases the signal power available to the receiver. Additionally, it provides protection against fading since several paths are unlikely to be subject to simultaneous deep fades. With suitable combining, this can improve the received signal-to-noise ratio, reduce fading and ease power control problems.

[0012] In conventional wireless communication systems, there is a significant frequency offset between the Node B and the WTRU due to imprecise oscillators used in the WTRU. This frequency offset, which translates into a phase shift over time, must be estimated and corrected in the WTRU or else a significant loss in performance will occur. There are several conventional algorithms used for differential detection to

estimate the phase shift in a constant velocity WTRU. The algorithms assume that the phase shifts between any two adjacent pilot symbols are constant over the observation window. The benefits of a Rake receiver sometimes are reduced because of complex algorithms required to perform frequency offset estimation and CWG which are processor and memory intensive, and consume valuable system resources.

[0013] Figure 2 shows three prior art simulated phase shift estimation algorithms 205, 210, 215, using the square root of the phase Mean Square Errors (MSEs) of three estimators at a Signal-to-Noise Ratio (SNR) of 0 dB.

[0014] A first prior art algorithm 205 supposes that $r_{k,j}$ is the j[th] despread pilot symbol at the kth slot. The phase shift (difference) θ between two adjacent pilot symbols can be estimated, $\hat{\theta}$, by Equation 1 as follows:

$$\hat{\theta} = angle\left\{\sum_{k=1}^{N_1}\sum_{j=1}^{N_2-1} r_{k,j+1} r_{k,j}^{*}\right\} \qquad \text{Equation 1}$$

where N_1 is the number of slots used for the phase shift estimation, and N_2 is the number of pilot symbols per slot used for the phase shift estimation.

[0015] A second prior art algorithm 210 estimates the phase difference of two pilot symbols that are separated by one symbol and divides it by two, and is expressed by Equation 2 as follows:

$$\hat{\theta} = \frac{1}{2} angle\left\{\sum_{k=1}^{N_1}\sum_{j=1}^{N_2-2} r_{k,j+2} r_{k,j}^{*}\right\} \qquad \text{Equation 2}$$

from the performance point of view, the larger the separation of two pilot symbols, the better the performance. But there is a limitation on the separation, which is the number of pilot symbols per slot. If the separation is too large, the system will not know how many phase rotations occurred, which will cause errors. Therefore the minimum number of pilot symbols is three per slot and two pilot symbols that are separated by more than one symbol cannot be used.

[0016] A third prior art algorithm 215 estimates the phase shift by using two pilot symbols separated by one slot. The phase shift over one slot estimation, $\hat{\theta}_0$, is shown by Equation 3 as follows:

$$\hat{\theta}_0 = angle\left\{\sum_{k=1}^{N_1-1}\sum_{j=1}^{N_2} r_{k+1,j}r_{k,j}^{*}\right\} \qquad \text{Equation 3}$$

where $-180^\circ < \hat{\theta}_0 \leq 180^\circ$. Since the phase shift over one slot is in the range of $-295^\circ \leq 10*\theta \leq 295^\circ$, there are ambiguities to estimating θ from $\hat{\theta}_0$. The values for $\hat{\theta}_0$ can be found in Table 1.

$$\hat{\theta} = \begin{cases} \frac{1}{10}\hat{\theta}_0 & |10*\theta| \leq 180^\circ \\ \frac{1}{10}(\hat{\theta}_0 + 360^\circ) & \theta > 0, \hat{\theta}_0 < 0 \\ \frac{1}{10}(\hat{\theta}_0 - 360^\circ) & \theta < 0, \hat{\theta}_0 > 0 \end{cases}$$

Table 1

[0017] The sign of θ is assumed to be known, and thus there is no ambiguity of prior art algorithm 215. The number of pilot symbols per slot is equal to 3. It is found that prior art algorithm 205 is the least effective and prior art algorithm 215 performs best, as expected.

[0018] The above prior art algorithms each have at least one problem. Prior art algorithm 215 outperforms prior art algorithms 205 and 210, but has phase ambiguity problems and cannot be used. Prior art algorithms 205 and 210 introduce a high noise variance.

[0019] Therefore new algorithms are needed which have better performance than the prior art algorithms 205 and 210, and which do not have the phase ambiguities of prior art algorithm 215. In addition, it would be desirable to these new algorithms produce complex weights needed for Rake reception that are less processor and memory intensive.

[0020] SUMMARY

[0021] A channel estimation method which reduces the strain on the resources of a Rake receiver using a CWG algorithm, and estimates and corrects frequency offset between Node B and the WTRU without the phase ambiguities or high noise variance of prior art algorithms.

[0022] In one embodiment, a non-adaptive algorithm is used to average blocks of pilot symbols from several slots. In another embodiment, an adaptive algorithm implements sliding window averaging or a recursive filter. Using a CWG algorithm reduces the memory and processor requirements of the Rake receiver.

[0023] BRIEF DESCRIPTION OF THE DRAWINGS

[0024] A more detailed understanding of the invention may be had from the following description of a preferred embodiment, to be understood in conjunction with the accompanying drawing wherein:

[0025] Figure 1 is a block diagram of a conventional Rake receiver;

[0026] Figure 2 illustrates the phase estimation performance of three prior art algorithms;

[0027] Figure 3A is a block diagram of a Rake receiver configured in accordance with the present invention;

[0028] Figure 3B is a block diagram showing the detailed configuration of the Rake Fingers used in the Rake receiver of Figure 3A in accordance with a preferred embodiment of the present invention;

[0029] Figure 4 illustrates the implementation block diagram of the estimation algorithm operating in accordance with one embodiment of the present invention;

[0030] Figure 5 graphically compares the simulation results obtained using prior art algorithms with an estimation algorithm for an SNR of 0dB using an N value equal to 32 in accordance with the present invention;

[0031] Figure 6 graphically compares the simulation results obtained using prior art algorithms with an estimation algorithm for an SNR of 0dB using an N value equal to 16 in accordance with the present invention;

[0032] Figure 7 shows an implementation of a cost function operating in accordance with the present invention; and

[0033] Figure 8 shows an implementation of a filter used in conjunction with a CWG algorithm operating in accordance with the present invention.

[0034] DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

[0035] The present invention is described with reference to the drawing figures wherein like numerals represent like elements throughout. The embodiments of the present invention can be applied to any Rake receiver, such as used in a wireless transmit/receive unit (WTRU) or a base station.

[0036] Hereafter, a WTRU includes but is not limited to a user equipment, mobile station, fixed or mobile subscriber unit, pager, or any other type of device capable of operating in a wireless environment. When referred to hereafter, a base station includes but is not limited to a base station, Node-B, site controller, access point or other interfacing device in a wireless environment. The preferred implementation is for use in receiving downlink common channels, although the features of the preferred embodiments can be applied to various implementations.

[0037] For illustrative purposes, the preferred embodiments are described in conjunction with a Dedicated Physical Common CHannel (DPCCH) the 3GPP wideband CDMA (W-CDMA) Frequency Division Duplex (FFD) mode. However, it can be applied in different channels in various wireless systems.

[0038] The DPCCH has pilot symbols, control data symbols, TPC and FBI symbols. Although the following description refers to pilot symbols, any reference signal can be used, such as a midamble sequence.

[0039] Figure 3A illustrates a block diagram of a Rake receiver 100 operating in accordance with the present invention. Rake receiver 100 includes an antenna 101, an

Automatic Gain Circuit (AGC) 103, an Analog to Digital Converter (ADC) 105, a Rake finger selector circuit 107, a plurality of delay elements 109_1, 109_2, 109_3, 109_4...109_N, a plurality of Rake fingers 111_1, 111_2, 111_3, 111_4...111_N, and the previously mentioned combiners 115 and 117. A spread modulation signal is received at the antenna 101 and is applied to the AGC 103, where the signal is amplified and conditioned. The output of the AGC 103 is input to the ADC 105 where the spread modulation signal is converted to a digital signal which is presented to the finger selector 107. The Rake finger selector 107 feeds the digital signal to the delay elements 109, the outputs of which are connected to a respective Rake finger 111. Each delay element 109 is set to delay the signal for a particular period of time that the delay provided by any of the other respective delay elements 109.

[0040] As shown in Figure 3A, each Rake finger 111 in Rake receiver 100 has two outputs, one being connected to combiner 115 and the other connected to combiner 117. The Rake receiver 100 outputs two data streams in parallel. Combiner 115 outputs a control information data stream, such as Transmit Power Control (TPC) and FeedBack Information (FBI) used for closed loop transmission diversity. Combiner 117 outputs a signal information data stream including detected data symbols.

[0041] Figure 3B illustrates an exemplary configuration of the Rake finger 111 used in Rake receiver 100 in accordance with a preferred embodiment of the present invention. The Rake receiver 100 and/or the Rake finger 111 may be incorporated into an integrated circuit (IC) or be configured in a circuit comprising a multitude of interconnecting components.

[0042] As shown in Figure 3B, the inventive Rake finger 111 of Rake receiver 300 includes despreaders 305, 310, a TBC/FBI bit processor 315, a frequency offset estimator 320, a complex weight gain generator 325, an alpha delay element 330 and a demodulator (or complex multiplier) 335. Each of despreaders 305, 310 receive an input signal from delay element 109. Additionally, despreader 305 receives a second input signal consisting of a pilot channel (e.g., DPCCH) despreading code and

despreader 310 receives a second input signal consisting of a data channel (e.g., dedicated physical data channel (DPDCH)) code.

[0043] The despreader 305 outputs despread pilot symbols which are applied to the inputs of TPC/FBI bit processor 315, frequency offset estimator 320 and complex weight gain generator 325. The output of frequency offset estimator 320 provides a second input to complex weight gain generator 325, which produces complex weight values by filtering the despread pilot (e.g., DPCCH) symbols received from the output of despreader 305.

[0044] The despreader 310 outputs despread data (e.g., DPDCH) symbols which are applied to the input of the alpha delay element 330 to ensure that data timing is aligned when the despread data symbols are multiplied by the complex weight values at demodulator 335. Thus, significant signal delays introduced by the frequency offset estimator 320 and complex weight gain generator 325 are eliminated.

[0045] The output of the demodulator 335 consists of weighted despread data symbols which are added to weighted despread data symbols from other Rake fingers 111 via combiner 117, as shown in Figure 3A. For example, if the data received at a particular Rake finger 111 does not have a strong correlation to the expected signal, the corresponding complex weight values provided by the complex weight gain generator 325 will approach zero. Thus, the demodulator 335 would essentially null out the signal at this particular Rake finger 111, and would apply little or no effect on the summation implemented at combiner 117.

[0046] On the other hand, if there is a strong correlation between the expected signal and the data received at the particular Rake finger 111, the complex weight values provided by the complex weight gain generator 325 would be relatively high. Thus, the output of demodulator 335 would be considerable at this particular Rake finger 111, thus having a significant effect on the summation implemented at combiner 117.

[0047] A time and frequency shift occurs using the frequency offset estimator 320 in response to an uplink transmission in a Universal Mobile Telecommunications

System (UMTS) frequency division duplex (FDD) system, e.g., due to the speed of a vehicle. For example, when an automobile is traveling at a constant 250 kms/hr (155 mph), a WTRU will experience a 0.613 ppm variation. Therefore, for this large frequency shift between the WTRU transmitter and node B receiver, there will be about 29.5 degree phase shift over one pilot symbol or over 256 chips. This degree of phase shift causes severe performance degradation within a Rake receiver. However, it is possible to estimate it and compensate for it in the CWG process since this phase shift is due to a constant frequency offset.

[0048] The CWG process required by the complex multiplier 335 puts a strain on the system resources of Rake receiver 300. For example, CWG algorithms requires large processor and memory usage. One way to reduce the strain on the system resources is to reduce size of the sliding window in the CWG algorithm and thereby reduce the memory and processor requirements.

[0049] Figure 4 shows the block diagram of a frequency offset estimator 320. The signal from the common channel despreader 305 is applied to the inputs of delay elements 405, 410, 415, and to a first input of each of complex multipliers 420, 425, 430, respectively. One illustrative value for the delays is one chip (T), two chips (2T) and ten chips (10T). The outputs of the delay elements 405, 410, 415 are applied to a second input of each of the complex multipliers 420, 425, 430, respectively. The output of the multipliers 420, 425, 430 are applied to summers 435, 440, 445, respectively. The outputs of the summers 435, 440, 445 are then applied to arithmetic calculator 450. The output of arithmetic calculator 450 is then used as an input to the complex weight gain generator 325.

[0050] The first sample w_1 estimates the phase difference of adjacent pilot symbols and is shown in Equation 4 as follows:

$$w_1 = \sum_{k=1}^{N_1} \sum_{j=1}^{N_2-1} r_{k,j+1} r_{k,j}^* \qquad \text{Equation 4}$$

where N_1 is the number of slots used for the phase shift estimation, N_2 is the number of pilot symbols per slot used for the phase shift estimation, and $r_{k,j}$ is the jth despread pilot symbol at kth slot.

[0051] The second sampling w_2 estimates the phase difference of two pilot symbols that are separated by one symbol and is shown in Equation 5 as follows:

$$w_2 = \sum_{k-1}^{N_1} \sum_{j-1}^{N_2-2} r_{k,j+2} r_{k,j} \qquad \text{Equation 5}$$

where N_1 is the number of slots used for the phase shift estimation, N_2 is the number of pilot symbols per slot used for the phase shift estimation, and $r_{k,j}$ is the jth despread pilot symbol at kth slot.

[0052] The third sample w_3 estimates the phase shift using two pilot symbols separated by one slot and is shown in Equation 6 as follows:

$$w_3 = \sum_{k=1}^{N_1-1} \sum_{j=1}^{N_2} r_{k+1,j} r_{k,j}^{*} \qquad \text{Equation 6}$$

where N_1 is the number of slots used for the phase shift estimation, N_2 is the number of pilot symbols per slot used for the phase shift estimation, and $r_{k,j}$ is the jth despread pilot symbol at the kth slot.

[0053] For a given phase estimation resolution Δ, let $N = \frac{\theta_{\max}}{\Delta}$; the result is shown in Equation 7 as follows:

$$F(k) = \mathrm{Re}\left(w_1 * e^{-j*k\Delta} + w_2 * e^{-j*k*\Delta*2} + w_3 * e^{-j*k*\Delta*10}\right) \qquad \text{Equation 7}$$

where, $-N \le k \le N$; search such a k_{opt} that maximizes $F(k)$, i.e., $F(k_{opt}) = \max_{-N \le k \le N} F(k)$, where $\mathrm{Re}(\bullet)$ is the real part, then $\hat{\theta} = k_{opt} * \Delta$.

[0054] Figure 5 shows the simulation results obtained with the above algorithm for an SNR of 0dB. The number of pilot symbols per slot is 3. The parameter N is set 32, which corresponds to a phase resolution of 0.92 degrees. An algorithm 500, operating in accordance with one embodiment of the present invention, performs much

better than the prior art algorithms 205 and 210, and performs almost identical to prior art algorithm 215 when the number of slots used for phase estimation is larger than 45. The algorithm 500 performs less effectively than the algorithm 215 for frames with smaller numbers of slots. However, the algorithm 215 uses the information of the sign of θ, which is not known in practice, and therefore it cannot be utilized in real life

[0055] Figure 6 shows similar simulation results as those in Figure 5, but instead uses an N value equal to 16, which corresponds to a phase resolution of 1.84 degrees. An algorithm 600, operating in accordance with another embodiment of the present invention, also outperforms prior art algorithms 205 and 210 and has results similar to the prior art algorithm 215 when the number of slots used for phase estimation is larger than 45.

[0056] The algorithm of the present invention can be reduced in complexity to maximize the cost function. Reducing one term at a time produces the following Equations 8, 9 and 10 as follows:

$$F_1(k) = \mathrm{Re}(w_2 * e^{-j*k*\Delta*2} + w_3 * e^{-j*k*\Delta*10}) \qquad \text{Equation 8}$$

$$F_2(k) = \mathrm{Re}(w_1 * e^{-j*k*\Delta} + w_3 * e^{-j*k*\Delta*10}) \qquad \text{Equation 9}$$

$$F_3(k) = \mathrm{Re}(w_1 * e^{-j*k*\Delta} + w_2 * e^{-j*k*\Delta*2}) \qquad \text{Equation 10}$$

[0057] The performances of these reduced-complexity algorithms have been simulated and it was found that the Equation 8 provides the smallest performance degradation. To further reduce complexity the calculation of the function $F(k)$ can take advantage of the symmetrical property of $F(k)$, which saves half of the multiplications normally required, which is shown in Equation 11 as follows:

$$\mathrm{Re}(w * e^{\mp j*k*\delta}) = w_r * \cos(k * \delta) \pm w_i * \sin(k * \delta) \qquad \text{Equation 11}$$

where $w = w_r + jw_i$.

[0058] Figure 7 shows an implementation 700 of the cost functions $\mathrm{Re}(w * e^{\mp j*k*\delta})$ of Equation 11. w_r is multiplied with $\cos(k * \delta)$ by multiplier 702. w_i is multiplied with sin $(k * \delta)$ by multiplier 704. The resulting products of multipliers 702 and 704

are added together by adder 706, producing $\mathrm{Re}(w * e^{-j*k*\delta})$. The result of multiplier 704 is subtracted from the result of multiplier 702 by subtractor 708, producing $\mathrm{Re}(w * e^{j*k*\delta})$.

[0059] The method of the present invention uses decision-feedback (DF) techniques implemented with a sliding window CWG algorithm implemented by complex weight gain generator 325 in Figure 3.

[0060] By using DF, both the pilot symbols and DPCCH data symbols can be used for channel estimation, reducing the required window size. Simulation results show that the DF approach can be used to shorten the window size of CWG algorithm.

[0061] The sliding window CWG algorithm of the present invention, first averages all pilot symbols in the sliding window as tentative channel gain estimation and then uses this tentative channel gain to demodulate the data symbols in the sliding window. After removing the demodulated information from the data symbols, the CWG algorithm sums all pilot symbols and de-rotated data symbols as final channel gain estimation.

[0062] The performance of sliding window CWG algorithm with DF was simulated for the Additive White Gaussian Noise (AWGN) channel with speeds of 3, 60, 120 and 250 km/hr. For comparison, the sliding window CWG algorithm without DF was also simulated, but its window size is twice as much as that used by the CWG with DF. It was found that the CWG with DF and with a K equal to five provides a tradeoff between the performance and implementation memory. However, when compared with CWG with a K equal to five without DF, the CWG with a K equal to five and with DF always provides the better performance.

[0063] Comparing a first CWG with a K equal to 10 without DF to a second CWG with a K equal to 5 with DF, the CWG with a K equal to 5 and DF provided similar performance for the AWGN channel and better performance in the Rayleigh fading channel at high speed of 250km/hr. The CWG with a K equal to 5 and DF exhibited performance derogation when the speed is slow.

[0064] To collect more data regarding the sliding window size and the performance degradation, more simulations have been made. Three sets of tests were performed, the first set the number of symbols for buffering at two, the second set is three and the third set at four. Each set simulated vehicle speeds of 3km/hr, 60km/hr, 120km/hr and 250km/hr, and utilized the AWGN and fading channels.

[0065] From the above simulations, it can be concluded that there is a tradeoff between performance and memory size. The larger the memory, the better the performance. Table 2 shows the performance and the shortened sliding window size The top horizontal row indicates the channel type. The left vertical column indicates whether it is a sliding window or a shortened sliding window algorithms and the K values. The values in the appropriate row and column indicate the root MSE (RMSE) of the different CWG algorithms at an SNR of 0dB.

	AWGN channel	Fading channel, 3km/hr	Fading channel, 60km/hr	Fading channel, 120km/hr	Fading channel, 250km/hr
Sliding Window K=10, No DF	0.30	0.30	0.30	0.30	0.47
Sliding Window K=5, with DF	0.30	0.35	0.35	0.35	0.40
Shortened sliding window K=2, with DF	0.45	0.48	0.48	0.48	0.52
Shortened sliding window K=3, with DF	0.38	0.42	0.42	0.42	0.43
Shortened sliding window K=4, with DF	0.35	0.38	0.38	0.38	0.43

Table 2

[0066] In addition to the method of deploying the complex weights using a DF algorithm, an embodiment of the present embodiment discloses a hybrid filter-predictor-type CWG algorithm with DF. This algorithm is targeted to achieve a compromise between the performance and implementation memory requirement. This algorithm utilizes future symbols.

[0067] The hybrid CWG algorithm first averages all pilot symbols in a sliding window as tentative channel gain estimation, where the sliding window is in the range -K1 to +K2, where K2 is the number of old symbols and K1 is the number of future symbols. For exemplary purposes, K1 is less than or equal to three.

[0068] The CWG uses tentative channel gain to demodulate the data symbols in the sliding window. First, all information is demodulated and removed from the data symbols. The CWG algorithm filters all pilot symbols and the de-rotated data symbols and outputs the filtered result as a final channel gain estimation.

[0069] Two filters are considered here, the first one has the linear function coefficients ($\{c_0, c_0+\Delta, \ldots, c_0+(K_1+K_2)\Delta\}$) and the second has the exponential coefficients ($\{\alpha^{K_1+K_2}, \ldots, \alpha, 1\}$).

[0070] The CWG algorithm essentially serves the function of a filter 800 shown in Figure 8. Filter 800, which may be implemented by a CWG algorithm running within complex weight gain generator 325, has K_1+K_2+1 filter coefficients as $\{C_{-K_1}, C_{-K_1+1}, \ldots, C_{-1}, C_0, \ldots, C_{K_2}\}$ and the phase shift from one symbol to the next one is Δ due to the frequency offset that is estimated by frequency offset estimator 320. The output of filter 800 is represented by Equation 12 as follows:

$$W_n = r_{n+K_1} C_{-K_1} e^{-jK_1\Delta} + \ldots + r_{n+1} C_{-1} e^{-j\Delta} + r_n C_0 + \ldots + r_{n-K_2} C_{K_2} e^{jK_2\Delta} \qquad \text{Equation 12}$$

[0071] As shown in Figure 8, filter 800 has K_1 leading taps for future symbols and K_2 lagging taps for past symbols. Each element "T" represents a delay box with one symbol delay and the DPDCH path is delayed by K_1 symbols for timing alignment. The input to the filter 800 is the despread symbol stream $r_{n+K_1}, \ldots, r_{n+1}, r_n, \ldots, r_{n-K_2}$. The

coefficients can be predetermined or adaptively changed. To achieve best effect, these coefficients should be changed according to vehicle speed.

[0072] Sliding window CWG algorithms with and without DF were simulated for comparison. For the non-DF sliding window CWG algorithm, its window size is twice as much as the CWG with DF. The channels are the AWGN and the fading channels. The speeds are 3km/hr, 60km/hr, 120km/hr and 250km/hr. The mean square errors (MSE) between the real channel gain and estimated channel gain are then calculated. The linear hybrid CWG algorithm uses the linear coefficients with $c_0 = \{0,0.2,0.4,0.6,0.8\}$ and $c_0 + (K_1 + K_2)\Delta = 1$.

[0073] The exponential hybrid CWG algorithm uses the exponential coefficients $\{\alpha^{K_1+K_2},\ldots,\alpha,1\}$ with an $\alpha = \{0.7,0.8,0.9\}$.

[0074] Several simulations of the various algorithms were then run while K2 and K1 were varied. The linear hybrid CWG algorithm using K2 equal to eight and K1 equal to two with the AWGN channel and fading channels with vehicle speed of 3km/hr, 60km/hr, 120km/hr and 250km/hr. K1 and K2 were incremented by one and the simulations were again performed.

[0075] The K2 was then set equal to eight and K1 equal to two and the exponential hybrid CWG algorithm was performed with the AWGN channel and fading channels with vehicle speed of 3km/hr, 60km/hr, 120km/hr and 250km/hr. K1 and K2 were incremented and the simulations were performed again.

[0076] Table 3 shows the performance comparison between different CWG algorithms. Two CWG algorithms are used for reference. One is the CWG algorithm with a K equal to ten (10-symbol data buffering) without DF, the other is the CWG algorithm with a K equal to five (5-symbol data buffering) with DF. The top horizontal row indicates the channel type. The left vertical column indicates whether it is a sliding window, the hybrid linear coefficients or the hybrid exponential coefficients. The values in the appropriate row and column indicate the RMSE of the different CWG algorithms at an SNR of 0dB.

	AWGN channel	Fading channel, 3km/hr	Fading channel, 60km/hr	Fading channel, 120km/hr	Fading channel, 250km/hr
Reference 1: Sliding Window K=10, No DF	0.30	0.30	0.30	0.30	0.47
Reference 2: Sliding Window K=5, with DF	0.30	0.35	0.35	0.35	0.40
Hybrid K1=2, linear coefficients	0.32	0.35	0.35	0.37	0.50
Hybrid K1=3, linear coefficients	0.31	0.34	0.34	0.36	0.45
Hybrid K1=2, Exponential coefficients	0.35	0.38	0.38	0.38	0.47
Hybrid K1=3, Exponential coefficients	0.35	0.37	0.37	0.38	0.45

Table 3

[0077] The following conclusions are based on the results from Table 3. The linear coefficient filter is better than the exponential coefficient filter. The linear coefficient filter with a K1 equal to three provides slightly improved performance than with a K1 equal to two. The CWG with a K equal to five is slightly better than the linear coefficient filter with a K1 equal to two.

[0078] While this invention has been particularly shown and described with reference to preferred embodiments, it will be understood by those skilled in the art that various changes in form and details may be made therein without departing from the scope of the invention as described hereinabove.

* * *